Password

Objective

Explore the tools available for implementing secure applications. Cryptographic hashing and encryption/decryption primitives are part of most programming language libraries. In this projectlet a personal password manager is developed.

With considerable flexibility in the way the user needs and wants are gathered, we will derive specifications. In formal environments, specifications are explicitly traced back to a need/want; similarly downstream design decisions may be traced to these specifications. This projectlet is an opportunity to implement such a disciplined approach.

It is not a goal to produce highly secure facility.

User needs and requirements

This projectlet is to fulfill the need of all users.

Id	Need/Requirement
UN-1	A password file is needed to store and retrieve passwords for a range of contexts e.g. websites.
UN-2	The password file should be encrypted with a password - which is required to inspect the contents.
UN-3	Even decrypted, the details including the password should not be stored in clear text.
UN-4	Contexts are identified by a name. Each user can have a username and password associated with each context.
UN-5	The utility should support creation of new contexts.
UN-6	The utility should support update of the password for a context.
UN-7	The utility should support the retrieval of the password for a context.
UN-8	For a context, the utility should support multiple user identities
UN-9	Any tampering of the password file should be detected and reported.

User wants

The following will enhance the appeal of the tool.

Id	Wants
UW-1	A convenient way to supply the password in a login context - without having to retype the password

Id	Wants
UW-2	When a password needs to be supplied, the utility shall prompt for it. The password is never to be entered in the command line.

Specifications

ID	Specification
DS-1	A utility **ppm** will be built to support the above.
DS-2	Switch -f to specify the password file name. The default is ~/.ppm/password.dat
DS-3	The password file is a comma/colon separated file.
DS-4	Command show (ppm show) to display the password for the context
DS-5	Command update (ppm update) to update the password for the context
DS-6	Command add (ppm add) to create a new entry
DS-7	Command list (ppm list) to show the entries in the password file
DS-8	First argument after the command provides the context
DS-9	Second argument after the command provides the username/identity
DS-10	The utility shall use the private ssl id for encryption/decryption operations.
DS-11	The switch -i:<private id file> specifies the private identity file. Default is ~/.ssh/id_rsa
DS-12	Command create creates the password file if it does not exist

Example usage